Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 5595–5602
Marseille, 11–16 May 2020

Reproducing Monolingual, Multilingual and Cross-Lingual CEFR Predictions

Yves Bestgen
LAST — UCLouvain
Place Cardinal Mercier, 10, 1348 Louvain-la-Neuve Belgium
yves.bestgen@uclouvain.be

Abstract

This study aims to reproduce the research of Vajjala and Rama (2018) which showed that it is possible to predict the quality of a text written by learners of a given language by means of a model built on the basis of texts written by learners of another language. These authors also pointed out that POStag and dependency n-grams were significantly more effective than text length and global linguistic indices frequently used for this kind of task. The analyses performed show that some important points of their code did not correspond to the explanations given in the paper. These analyses confirm the possibility to use syntactic n-gram features in cross-lingual experiments to categorize texts according to their CEFR level (Common European Framework of Reference for Languages). However, text length and some classical indexes of readability are much more effective in the monolingual and the multilingual experiments than what Vajjala and Rama concluded and are even the best performing features when the cross-lingual task is seen as a regression problem. This study emphasized the importance for reproducibility of setting explicitly the reading order of the instances when using a K-fold CV procedure and, more generally, the need to properly randomize these instances before. It also evaluates a two-step procedure to determine the degree of statistical significance of the differences observed in a K-fold cross-validation schema and argues against the use of a Bonferroni-type correction in this context.

Keywords: automated essay scoring, k-fold cross-validation, statistical significance test

1. Introduction

Automated essay scoring (AES) is one of the main application areas of natural language processing in education and teaching as evidenced by the commercial systems available on the market for many years such as the Intelligent Essay Assessor, eRater or IntelliMetric (Shermis and Hammer, 2013). Used most often as a second assessment in addition to a human evaluator, these systems facilitate the deployment of standardized tests by reducing the necessary human resources (Williamson et al., 2012). Such systems are particularly useful in the field of foreign language learning where being able to write high-quality texts is of most importance (Bestgen, 2019).

The state-of-the-art approach in this area relies on linguistic clues potentially correlated with the text quality such as the presence of typographical, orthographic and syntactic errors, detected among others on the basis of n-grams, but also the length of the text, the presence of very short sentences, the absence of an introduction or a conclusion (Williamson et al., 2012). Training materials, consisting of at least several hundred texts evaluated by human experts, are used to establish the most effective model for predicting these human evaluations. This model is then applied to new texts.

Most of the works in this area have been focused on English as a second language where the needs are the most obvious (Condon, 2013; Weigle, 2013). However, although knowing a lingua franca is important, being able to integrate oneself into another culture by communicating in its language is also important. Recently, studies have been conducted to develop automatic quality assessment systems for other languages such as German or Chinese (Hancke and Meurers, 2013; Zhang et al., 2016). These works benefit from the studies conducted in English, but they nevertheless require significant resources, especially to collect the large sets of graded texts necessary for developing and testing the supervised learning procedures that are the state-of-the-art in this area (Ramineni and Williamson, 2013).

In this context, Vajjala and Rama's (2018) recent study is particularly important. Not only does it focus on three non-English languages (i.e., German, Italian and Czech), but it also undertakes what seems to be the first attempts at devising a multilingual classifier, as well as a cross-lingual classifier by learning a predictive model on one language and testing it on another. The main objective of the authors is therefore nothing less than to answer the question "Is there a universal model for language proficiency classification?". In their study, Vajjala and Rama (2018) compared the performances of words, POStag and syntactic dependencies n-grams with those of global linguistic features, such as text length and readability indices, to predict the CEFR category in the three languages. They observed that the n-grams outperformed the other features and that they achieved a satisfactory level of performance in multilingual and cross-lingual tasks.

These results are full of opportunities, but also largely unexpected according to the authors themselves (Vajjala and Rama, 2018, p. 147). The low level of performance of text length is also at odds with one of the most virulent criticisms made to AES: being only a measure of this characteristic (Perelman, 2014). The inclusion of this study in REPROLANG 2020 should allow a better understanding of these results by promoting their reproduction by other researchers and, therefore, by multiplying the points of view on the data, procedures and conclusions of the original study. Participating in this effort is the main objective of this research. The following section briefly describes the original study. Then, the major difficulties encountered during the reproduction attempts are discussed. Finally, the analyses conducted to determine whether or not it is possi-

ble to reproduce Vajjala and Rama's (2018) conclusions are presented as well as the results obtained.

2. Brief Description of the Original Study

2.1. Materials

The material used in Vajjala and Rama (2018) was extracted from the MERLIN corpus (Boyd et al., 2014), which can be freely redistributed. It contained 2,267 texts, such as letters or e-mails, written by learners of German (DE), Italian (IT), and Czech (CZ) as a second language. These texts had been graded according to the Common European Framework of Reference (CEFR, 2001), which includes six categories - A1, A2, B1, B2, C1, C2 - from the beginner level to the advanced level.

Table 1 shows the distribution of the texts in each language according to their CEFR level. This table is identical to Table 1 of Vajjala and Rama (2018) except that they report 393 texts at the B1 level for Italian. The results provided by these authors indicate that their analyses were also conducted on 394 B1 texts in Italian[1].

Very large disparities between the three languages are observed. For example, no text in Italian is higher than the B1 level while there are 335 (32%) such texts in German and 81 (18%) in Czech. There is also no text at the A1 level in Czech, contrary to what is observed in German and Italian. These differences in distribution could affect the predictive model performance, especially in the cross-lingual experiments.

CEFR	DE	IT	CZ
A1	57	29	0
A2	306	381	188
B1	331	394	165
B2	293	0	81
C1	42	0	0
Total	1029	804	434

Table 1: Distribution of the texts according to the CEFR level and language

2.2. Methods

2.2.1. Feature sets

The three main sets of features (or conditions) are composed of n-grams (*n* ranging from 1 to 5) of words, POStag and dependency relationships obtained using the UDPipe parser (Straka et al., 2016) trained on Universal Dependencies treebanks (Nivre et al., 2016). Vajjala and Rama (2018) also employed a set of linguistic features borrowed from classical AES studies (called Domain features) that includes text length, lexical richness indexes and, for German and Italian, statistics about the number of errors, obtained through LanguageTool[2]. Each of the three types of n-grams was also combined with these Domain features, producing three additional sets. Finally, they used word and character embeddings derived exclusively from the learning materials. The detailed procedure is described in the original paper.

2.2.2. Supervised Learning

To learn the models for all types of features except embeddings, Vajjala and Rama (2018) compared three standard supervised learning procedures : Logistic Regression, Random Forests, and Support Vector Machines, as implemented in scikit-learn (Pedregosa et al. 2011). For word and char embeddings, they used neural network models implemented using Keras (Chollet et al., 2015) and TensorFlow (Abadi et al., 2015).

2.2.3. Evaluation

Vajjala and Rama (2018) employed weighted F1-scores to gauge the performance of their models because of the unbalanced class distribution.

2.3. Experiments

Three experiments were run by varying the learning and testing materials. The first experiment (monolingual) is based on the usual approach of learning and testing a model on the same language using a stratified K-fold cross-validation (CV) procedure (with $K = 10$). In the second experiment (multilingual), the texts of the three languages were aggregated into one set and analysed using the same cross-validation procedure. The authors compared the results obtained when features indicating the language in which a text is written were present (Lang (+)) or not (Lang (-)). Finally, the German set of texts, the largest sample, was used in the third experiment (cross-lingual) to independently predict the level of texts in Italian and Czech, without the need for a cross-validation procedure since the learning and testing materials did not overlap.

3. Main Difficulties with the Reproduction

The main objective of this study is to reproduce as closely as possible Vajjala and Rama (2018). A priori, it seems like an extremely simple task since the authors have provided, not only the complete datasets, but also the Python code they used, including the parameters and seeds for the random generators, and even the logs of the analyses carried out[3]. It should be noted however that the version of Python and the modules used is not known. No difficulty was encountered in the preprocessing steps. However, this was not the case with the following ones.

3.1. Problems with Cross-Validation

The first reproduction trials, conducted under macOS, to facilitate developments, and in a Docker image, in order to be able to comply later with the REPROLANG 2020 requirements[4], produced unexpected results. While both versions

[1] See the files in the folder *results/final_results_for_paper/* provided by Vajjala and Rama (Note 3).

[2] `https://languagetool.org/`

[3] Available at `https://github.com/nishkalavallabhi/UniversalCEFRScoring`

[4] This Docker image is publicly available at `https://gitlab.com/CrVa/reprolang2020_cefr` (commit: 6b410f00, tag: V1.0). This image is based on Daniel Rapp's Docker file (https://github.com/rappdw/docker-java-

ran the same code on the same version of Python (and modules), using the same parameters and seeds for the random generators provided in the original code, the Docker version produces results similar to those reported in the original study while the macOS version produces very different results from both the Docker version and the original results. Table 2 presents the F1-scores[5] for the multilingual analysis obtained under Docker and under macOS and the deviations from the values reported in the original study (see Table 3 in Vajjala and Rama (2018)).
The Docker vs. macOS comparison highlights very important differences, especially in the case of the Word n-grams. A thorough analysis of the code shows that these differences have their origin in two factors:

- The fact that the texts to be analysed are in different files whose reading order, and therefore the order of the instances in the material to be analysed, is dependent on the version used;
- The use of scikit-learn (version $\leq$ 0.21.3) *StratifiedKFold* function with a *random_state* parameter, but without setting the *shuffle* parameter to *True*, which leads to an absence of randomization of the instances before the fold assignment.

These two factors explain the observed differences since the content of the folds is not the same depending on the version. This explanation is confirmed by the following experiment: if the code is modified so that the texts are read in the same order under Docker and macOS, exactly the same values in all the conditions are obtained[6].
These problems could have had a large impact on the study conclusions. While the results for the Docker version are relatively close to those reported in the original study, macOS results are relatively far to the point that the Word n-gram condition, which in the original study and in the Docker version is among the most useful features, is much less effective than the POStag features in the macOS version. How could we explain such differences? A first hypothesis is that the CV results depend strongly on the content of the folds and are therefore not stable. Another hypothesis is that the arbitrary initial order of the texts has led to large differences. This second hypothesis is supported by an important consequence of the absence of randomization before the fold assignment. The proportion of texts of each language in the folds is highly variable, some test folds containing only texts written in one language[7]. To try to determine which one of these hypotheses better explains these differences, section 4.1 reports an analysis in which the order of the texts is randomly switched ten times so as to obtain ten different CVs and thus be able to evaluate the stability of the results.

	Lang (-)		Lang (+)	
	Docker	macOS	Docker	macOS
Baseline	0.579l	0.574l	0.625l	0.617l
	+0.151	+0.146	N/A	N/A
Word	0.720	0.606	0.723	0.607
	-0.001	-0.115	+0.004	-0.112
POStag	0.722	0.680	0.725	0.681
	-0.004	-0.046	+0.001	-0.043
Dependency	0.699	0.651	0.699	0.653
	-0.004	-0.052	+0.006	-0.040
Domain	0.635	0.597	0.696	0.647
	+0.186	+0.148	+0.225	+0.176
Word + char	0.692	0.668	0.681	0.659
	-0.001	-0.025	-0.008	-0.030

Table 2: Docker and macOS weighted F1-scores and deviations from the values reported in the original study (second line) for the multilingual experiment

3.2. Problems with the F1-Scores

Table 2 also shows that the F1-scores for the Baseline and Domain conditions are significantly lower in the original study than in the two reproduction trials. The reason for this discrepancy lies in the use in the original code of a macro-averaged F1-score for these two conditions instead of the weighted F1-score indicated in the paper and used for the other conditions. This greatly modifies several conclusions of the study.
One last difficulty, encountered in Vajjala and Rama's code, must be mentioned, even if its impact on the results is weak. In all analyses, except those based on the neural models, a global F1-score is calculated, taking into account the predicted and observed values of all folds while the F1-scores for the neural network models are the average of the F1-scores calculated separately for each fold. Further analyses confirmed that both procedures produced very similar results. In the following, all the results presented are based on the overall F1-score.

4. Analyses and Results

4.1. The Mono- and Multilingual Experiments

These two experiments are discussed simultaneously because both rely on the cross-validation procedure that

python/blob/master/Dockerfile). The datasets necessary to reproduce the results reported here are available at `https://sites.google.com/site/byresearchoa/z/doc/input.tar.gz?attredirects=0&d=1`. This file MD5 is 691fb9899b69d6b96bb2f7ced3439895.

[5]As in Vajjala and Rama (2018), the reported values are those produced by the best-performing classifier, almost always the Random Forest classifier except when *l* is suffixed to the F1-score, which means that the Logistic Regression model was the best.

[6]It must be noted that the neural network models do not lead to exactly the same values in different runs

[7]Two factors cause this disparity. First, the initial order of the texts submitted to the (Stratified) K-Fold procedure is partially systematic since all the DE texts precede the IT texts that precede the CZ texts. Then, the K-Fold (Stratified) procedure always uses the first texts submitted to it (after or not a randomisation step) to fill in the first test fold, the following texts to fill the second one, and so on. It follows that the first test folds contain for each CEFR level (the stratification criterion) a larger proportion of DE texts while the corresponding learning material contains a larger proportion of IT and CZ texts while the reverse is true for the last folds.

	DE			IT			CZ		
Condition	Mean	Range	Dev.	Mean	Range	Dev.	Mean	Range	Dev.
Baseline	0.627	0.007	+0.130	0.808	0.004	+0.230	0.628	0.017	+0.041
Word n-grams	0.660	0.016	-0.006	**0.832**	0.014	+0.005	**0.732**	0.033	+0.011
POStag n-grams	0.677	0.016	+0.014	0.820	0.011	-0.005	0.699	0.025	0.000
Dep. n-grams	0.658	0.013	-0.005	0.808	0.013	-0.005	0.723	0.032	+0.019
Domain features	0.647	0.014	+0.114	0.818	0.010	+0.165	0.682	0.026	+0.019
Word + Domain	0.676	0.020	-0.010	0.831	0.018	-0.006	0.722	0.044	-0.012
POStag + Domain	**0.680**	0.020	-0.006	0.819	0.016	-0.006	0.713	0.046	+0.004
Dep. + Domain	0.675	0.023	-0.007	0.810	0.021	+0.004	0.720	0.053	+0.008
Word embeddings	0.639	0.022	-0.007	0.810	0.021	+0.016	0.660	0.047	+0.035

Table 3: Weighted F1-score (mean, range and deviations from the values reported in the original study) for the monolingual experiment

caused specific problems during the reproduction. The following section describes the statistical techniques used to try to determine whether it is possible to reproduce Vajjala and Rama's (2018) results and conclusions.

4.1.1. Statistical Techniques

The results presented in this section were obtained by blocking the reading order of the texts and activating the *Shuffle* option in the (Stratified) K-Fold CV function to avoid the problems reported above. In order to evaluate the stability of the results according to the composition of the folds, ten CVs, based on different random seeds, were carried out.

Undertaking several CVs necessarily highlights some variability that can affect the conclusions. It is therefore necessary to be able to determine whether the differences between the average values across the folds are sufficiently important to be considered reproducible. In fact, this problem arises even when only one (K-fold) cross-validation is performed as it is the case in Vajjala and Rama (2018). How do we decide whether the difference observed between two conditions is large enough to be considered reproducible? This is one of the functions of the statistical significance tests. For over twenty years this question has been the focus of attention in computational linguistics, especially because of the lack of independence between the different CVs (e.g., Bouckaert and Frank, 2004; Dietterich, 1998; Dror et al., 2018; Nadeau and Bengio, 2003). On the basis of these works, the following two-step procedure was used. First, it was necessary to determine whether the difference between two conditions in each CV is large enough to claim that it is reproducible. Then it was necessary to make a decision on the basis of all the conclusions obtained in the 10 CVs. Let us consider these two steps successively.

For each CV in each condition, an F1-score (or any other relevant measure) can be calculated on the basis of the predicted and correct values for each text. As identical random seeds are used in all conditions, the predictions for a given CV in the different conditions are obtained on the basis of the same random distribution of the instances and are therefore true repeated measures (only the predictive model changes between the conditions). The F1-scores obtained in two different conditions for the same random partition can thus be compared by means of a randomization test for repeated measures, the Fisher-Pitman test (Berry et al., 2002; Bestgen, 2017; Dror et al., 2018; Neuhauser and Manly, 2004), using 2,000 permutations. This test is a close relative of the classical Student's t-test which is often seen as problematic in NLP because it is based on a postulate of normality difficult to sustain. Fisher-Pitman's test is identical to the Wilcoxon-Mann-Whitney's nonparametric test except that the latter is calculated on ranked data while the former is based on raw data.

This first step ends, for each comparison, with ten p-values corresponding to ten differences between the F1-score of the two compared conditions. How can a general conclusion about these results be drawn? As these tests are not independent, they cannot be summarized by using, for example, a binomial test. On the other hand, the following relevant index can be computed: the number of CVs for which the probability obtained is less than the classical α of 0.05. This index informs about the chance that a CV, based on a randomly selected seed, produces a statistically significant result. Using a Bonferroni-type procedure for reducing the α threshold according to the number of CVs conducted is not justified here because it would penalyze more strongly studies that undertake more CVs and are thus more informative about the reproducibility of a difference.

The index mentioned above is based on an inferential test. There is however a simpler way to answer the question of whether an observed difference is reproducible when several CVs are performed. It consists of counting the number of CVs for which the best score is obtained in the condition which is on average the best. This index informs about the chances that the best score would be obtained by the same condition in any CV. In the following, these two indices will be compared.

The main difficulty encountered by this approach in reproducing Vajjala and Rama (2018) stems from their use of several supervised learning procedures and the presentation, for reasons of space availability, of the results of the best procedure for each condition. When several CVs are run, there is no guarantee that the same procedure will always be the most effective. Since Vajjala and Rama (2018) observed in their experiments that the Random Forests were generally the most effective, this procedure was used for almost all conditions. The only exception was the Base-

line condition which is very different since it is based on a single feature. For this condition, Logistic Regression was used because Vajjala and Rama (2018) observed that it was the most effective.

4.1.2. Results

Table 3 presents the mean F1-scores obtained on the basis of the ten CVs as well as the ranges (differences between the highest and the smallest values) for the monolingual experiments (Table 2 in Vajjala and Rama (2018)). It also gives the difference between each mean value and the corresponding value reported in Vajjala and Rama (2018) when available, a "+" indicating that the reproduction led to the highest value. For every condition, including the Baseline and the Domain conditions, the F1-scores reported were obtained using the weighted average.

This table confirms that Vajjala and Rama (2018) did not use the weighted F1-score for the Baseline and Domain conditions. This strongly modifies the conclusions since the performances of the Baseline are no longer lower than those of the other conditions by almost 30% in German and Italian, but by less than 10%. Similarly, the Domain features are no longer strongly less effective than the other conditions.

It is also noted that the range is rather low in all the conditions except in Czech, the smaller corpus. With the exception of the Baseline and Domain conditions, the differences between the average values obtained during the reproduction and those reported in the original study are also small and broadly compatible with the ranges.

In order to go deeper into the comparison of the different types of features, the condition that obtained the highest performance for each language (bolded in Table 3) was compared to all others using the statistics described above[8] and the results are reported in Table 4. The figures correspond to the number of CVs for which the Fisher-Pitman test is significant (Sig.) at an α level of 0.05 and for which the difference between the two conditions is not inverted (¬Inv.). In 14 out of 24 comparisons, more than half of the CVs made were statistically nonsignificant. As a result, it's hard to argue that many conditions are significantly less effective than the best ones. This result puts the differences between the means into perspective. Regarding the second index, it is very reassuring to note that inversions of the best score somewhat rarely occur (in 7 out of 32 comparisons).

The most interesting result to evaluate the two-step approach is that there is only one case (Word + Domain in IT) out of 24 where a statistically significant CV is observed while there is at least an inversion. It follows that, in all other cases where there is at least one inversion, none of the CVs are statistically significant. This observation suggests that performing a Fisher-Pitman's test on one CV is already highly informative about the existence of a real difference between the two compared conditions.

[8] A Bonferroni-type correction, recommended to take into account the pairwise comparisons between the conditions made for each learner corpus, is not used here because it would have made these analyses even more unfavorable to Vajjala and Rama (2018) since the significance thresholds would have been much smaller.

Condition	Mean	Sig.	¬Inv.
DE: POStag + Domain = 0.680			
POStag n-grams	0.677	0	6
Word + Domain	0.676	0	6
Dep. + Domain	0.675	0	6
Word n-grams	0.660	1	10
Dep. n-grams	0.658	6	10
Domain features	0.647	9	10
Word embeddings	0.639	10	10
Baseline	0.627	10	10
IT: Word n-grams = 0.832			
Word + Domain	0.831	1	4
POStag n-grams	0.820	0	10
POStag + Domain	0.819	1	10
Domain features	0.818	1	10
Dep. + Domain	0.810	3	10
Word embeddings	0.810	10	10
Baseline	0.808	6	10
Dep. n-grams	0.808	7	10
CZ: Word n-grams = 0.732			
Dep. n-grams	0.723	0	6
Word + Domain	0.722	0	8
Dep. + Domain	0.720	0	7
POStag + Domain	0.713	1	10
POStag n-grams	0.699	1	10
Domain features	0.682	6	10
Word embeddings	0.660	10	10
Baseline	0.628	10	10

Table 4: Number of CVs for which the difference between the best condition and each of the others is reproduced for the monolingual experiment

Table 5 presents the CV results for the multilingual experiment (Table 3 in Vajjala and Rama (2018)). It shows that the ranges are much smaller than the differences observed between the Docker and macOS versions reported in Table 2, confirming that the macOS order, which has not been randomized, arbitrarily affected the results. Again, the values that are the most distant from those reported in the original study comes from the Baseline and Domain conditions. While the Baseline remains very ineffective, Domain is very close to the other conditions when features encoding the text language are provided to the supervised learning procedure. The inferential tests confirm this conclusion, as well as the observations reported above, and are not reported here.

4.2. The Cross-Lingual Experiment

4.2.1. Statistical Techniques

The cross-linguistic experiment conducted by Vajjala and Rama (2018) deserves special attention because of its originality and scope. It differs from the previous experiments by using, as expected, the weighted F1-score for the Baseline and Domain conditions. As it is not based on a CV, the authors' original approach, which retains the highest F1-scores of the three supervised learning techniques, can

	Lang (-)			Lang (+)		
Condition	Mean	Range	Dev.	Mean	Range	Dev.
Baseline	0.590	0.006	+0.162	0.640	0.006	N/A
Word n-grams	**0.732**	0.011	+0.011	**0.733**	0.009	+0.014
POStag n-grams	0.728	0.007	+0.002	0.729	0.008	+0.005
Dep. n-grams	0.711	0.010	+0.008	0.712	0.009	+0.018
Domain features	0.646	0.009	+0.197	0.711	0.011	+0.240
Word+Char embed.	0.689	0.009	-0.004	0.684	0.006	-0.005

Table 5: Weighted F1-score (mean, range and deviations from the values reported in the original study) for the multilingual experiment

be used even if it seems like using an oracle. These F1-scores were compared for all pairs of conditions taken two by two using the Fisher-Pittman randomization test, using 2,000 permutations, with the significance level set at 0.05, so without Bonferroni correction.

In this experiment, the texts written in German are used to predict separately the CEFR level of the Italian and Czech texts. However, as shown in Table 1, there are very large disparities between the three languages regarding the number of texts classified in each CEFR level. These differences in distribution might affect the model performance when the evaluation is based on the proportion of correct classifications. This problem can be circumvented by taking into account the fact that the CEFR categories form a scale ranging from the lowest level to the highest. The prediction task can then be modelled as a regression task, as emphasized by Vajjala and Rama (2018, p. 149), an extremely common approach in AES (Attali and Burstein, 2006; Ramineni and Williamson, 2013). This makes all the more sense in the case of the cross-lingual experiment because Vajjala and Rama (2018) observed "that most of the misclassification occurs only between adjacent levels of proficiency" (p. 151). These authors did not report the results for the regression models but indicate that they performed very well. These analyses were therefore carried out using three regression procedures proposed by these authors in their code, namely Linear Regression, Random Forest Regressor, and Gradient Boosting Regressor. These procedures were applied exactly to the same features as those used in the classification task and the Pearson correlation coefficient was used as the evaluation metric (similar results were obtained with Spearman *rho*).

4.2.2. Results

The results for the classification task are given in Table 6 (Table 4 in Vajjala and Rama (2018)). The values obtained are very close to those reported in the original study, and even identical in the case of the Baseline, except for Domain in Czech[9]. The inferential tests show that for Italian, POStag is significantly better than all other conditions and Baseline significantly worse than all other conditions. There is no significant difference between Dependency and Domain. For Czech, POStag and Dependency are better than Baseline and Domain while the differences within these pairs of conditions are not statistically significant. These results confirm all the conclusions of Vajjala and Rama (2018) but one, namely that "In the case of Italian, the domain features yield similar results to monolingual results suggesting that there are some possible universal patterns of language use in the progression towards language proficiency" (p. 151). This conclusion is no longer correct when the weighted F1-score is used in the monolingual analysis since the value obtained is 0.818, which is significantly higher than 0.625.

Condition	Test:IT	Test:CZ
Baseline	0.553l	0.487l
	0.000	0.000
POStag n-grams	**0.743**	**0.673**
	-0.015	+0.024
Dependency n-grams	0.648	0.670
	+0.024	+0.017
Domain features	0.625l	0.562s
	+0.005	+0.087

Table 6: Classification results (mean F1-score and deviations from the values reported in the original study) for the cross-lingual experiment

Condition	Test:IT	Test:CZ
Baseline	0.729	0.698
POStag n-grams	0.710	0.658r
Dependency n-grams	0.689	0.644
Domain features	**0.745**	**0.728**

Table 7: Regression results (Pearson *r*) for the cross-lingual experiment

The results for the regression task are very different (Table 7[10]). The most successful condition is Domain which is sig-

[9]It is not easy to explain the origin of this difference. It happens in a condition where the performance obtained by Vajjala and Rama (2018) is very low (0.475) and by means of the learning technique (Support Vector Machines, marked by an s in the Table) which in almost all the analyses does not converge.

[10]The reported values are those produced by the best-performing regressor, almost always the Gradient Boosting Regressor except when *r* is suffixed to the value, which means that the Random Forest Regressor was the best.

nificantly better than POStag and Dependency in both languages but not than Baseline in IT, Baseline being significantly better than Dependency in both languages and better than POStag in CZ. These results suggest that, treated as a regression task, the prediction of the CEFR level by a cross-linguistic approach is not much more than what the length of the text allows.

5. Discussion and Conclusion

In attempting to reproduce Vajjala and Rama (2018), my first goal was to better understand the results reported by these authors. The analyses support their conclusion that it is possible to use features like POStag or dependency n-grams to learn a predictive model on one language and then use it to categorize texts written in another language. However, the complementary analyses suggest that the number of words in each text and some classical indexes of readability are more effective when the task is seen as a regression problem. The path opened by Vajjala and Rama seems promising, but further studies that take into account the distribution of the CEFR levels in the learning and testing materials are necessary. In these studies, the impact of the average CEFR level of the texts in the materials should receive careful attention. The results reported above indicate that the effectiveness of the baseline, the text length, is very variable according to the corpus, much better in Italian than in Czech. Texts in Italian are of a lower overall level and text length is known to be a particularly effective index to distinguish between beginner and intermediate learners (Chodorow and Burstein, 2004). Having a corpus in which all CEFR levels are represented in equal proportions should allow the development of more effective models. The results presented would have been more complete if I had performed a qualitative analysis of the most important errors produced by the different models in the different languages. Unfortunately, my lack of knowledge of these three languages made such an analysis impracticable.

The present study was also a reproduction exercise conducted on Vajjala and Rama (2018). In the context of REPROLANG 2020, it may seem surprising that this objective is not mentioned in the first place. The reason is that, although developing good practices for reproducing scientific studies is very important, the exercise of reproduction in itself is especially interesting when the original study can be considered as a landmark, which is the case for Vajjala and Rama (2018) as explained in the introduction. When this is less true, it is unclear why a given study is more worth reproducing than each of the numerous other studies published each year in all areas of NLP.

It was expected that reproducing Vajjala and Rama (2018) would be an extremely simple task since the authors provided the complete data, the code and all the parameters. However, this exercise showed that several important points of the code did not correspond to the explanations given in the paper. Such a situation is most likely the result of a research being rarely fully planned in advance. The development phase often leads to many attempts that are dropped when they do not prove useful, but which may leave traces in the code. This is a mistake I made myself (Bestgen, 2018) and, although the remedy seems simple (i.e., clean up, simplify and check the final code as much as possible), it seems very difficult to guarantee that such an error did not happen. Reproducing also allows a different view on the decisions made during the development phase (for instance, the length of the n-grams). These decisions are not always conducted on a sample independent of the one which will be used during the test phase, making overfit likely.

Reproducing is therefore useful even when the authors provide all the necessary elements. Carefully analyzing the code (and nothing proves that I did not miss some important points during this analysis) is even more fundamental. This is particularly the case when the code is based on functions borrowed from modules that are provided by other researchers (see for instance the discussion above of the parameters *random_state* and *shuffle* in scikit-learn (version $\leq$ 0.21.3[11]) *StratifiedKFold* function). It follows that making the code available in the form of a Docker image as it was required for REPROLANG 2020 is useful in order to ensure that the results can be reproduced, but it does not guarantee that these results correspond to the explanations given in the paper. On the other hand, providing a Docker image makes it possible to find the versions of the programs and modules used, often necessary for reproducing exactly a study.

A final feature of this study is the attention paid to the use of a K-fold cross-validation procedure. First, this study emphasized the importance for reproducibility of setting explicitly the reading order of the instances when using a K-fold CV procedure and, more generally, the need to properly randomize these instances before. Second, a two-step procedure to determine the degree of statistical significance of the differences observed in a K-fold cross-validation schema was evaluated. The results suggest that applying the Fisher-Pittman's test to a single CV is already very informative as long as an adequate randomization procedure is used when constructing the learning and validation sets. This observation must be reproduced in other studies because the data and especially the number of available instances is obviously a factor to be taken into account, as confirmed by the observation that the ranges in Table 3 are much larger in the smallest corpus. In addition, it goes without saying that the use of several CV makes it possible to be even more confident regarding the existence of a real difference.

6. Acknowledgements

The author is a Research Associate of the Fonds de la Recherche Scientifique - FNRS. He would like to thank the REPROLANG 2020 CLARIN Team for the invaluable help in building the Docker image on GitLab and the reviewers for their very constructive comments.

7. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M.,

[11] This issue should be corrected from version 0.24 of sklearn.

Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Attali, Y. and Burstein, J. (2006). Automated essay scoring with e-Rater v.2.0. *Journal of Technology, Learning, and Assessment*, 4(3).

Berry, K. J., Mielke, P. W., and Mielke, H. W. (2002). The Fisher-Pitman permutation test: an attractive alternative to the F test. *Psychological Reports*, 90:495–502.

Bestgen, Y. (2017). Getting rid of the Chi-square and Log-likelihood tests for analysing vocabulary differences between corpora. *Quaderns de Filologia: Estudis Linguistics*, 22:33–56.

Bestgen, Y. (2018). Predicting second language learner successes and mistakes by means of conjunctive features. In *Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications, NAACL-HLT 2018*, pages 349–355.

Bestgen, Y. (2019). Evaluation de textes en anglais langue étrangère et séries phraséologiques : comparaison de deux procédures automatiques librement accessibles. *Revue Française de Linguistique Appliquée*, 24:81–94.

Bouckaert, R. R. and Frank, E. (2004). Evaluating the replicability of significance tests for comparing learning algorithms. In Honghua Dai, et al., editors, *PAKDD*, volume 3056 of *Lecture Notes in Computer Science*, pages 3–12. Springer.

Boyd, A., Hana, J., Nicolas, L., Meurers, D., Wisniewski, K., Abel, A., Schöne, K., Štindlová, B., and Vettori, C. (2014). The MERLIN corpus: Learner language and the CEFR. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1281–1288, Reykjavik, Iceland. European Language Resources Association (ELRA).

Burstein, J., Chodorow, M., and Leacock, C. (2004). Automated essay evaluation: The Criterion Online Writing Service. *AI Magazine*, 25:27–36.

Chodorow, M. and Burstein, J. (2004). Beyond essay length: evaluating e-rater's performance on TOEFL essays. Technical report, Educational Testing Service.

Chollet, F. et al. (2015). Keras. `https://keras.io`.

Condon, W. (2013). Large-scale assessment, locally-developed measures, and automated scoring of essays: Fishing for red herrings? *Assessing Writing*, 18(1):100–108.

Council of Europe. (2001). *Common European Framework of Reference for Languages: Learning, Teaching, Assessment.* Cambridge University Press.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923.

Dror, R., Baumer, G., Shlomov, S., and Reichart, R. (2018). The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.

Hancke, J. and Meurers, D. (2013). Exploring CEFR classification for German based on rich linguistic modeling. In *Paper presented at Learner Corpus Research Conference (LCR 2013)*.

Nadeau, C. and Bengio, Y. (2003). Inference for the generalization error. *Machine Learning*, 52(3):239–281.

Neuhauser, M. and Manly, B. F. J. (2004). The Fisher-Pitman permutation test when testing for differences in mean and variance. *Psychological Reports*, 94:189–194.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

Perelman, L. (2014). When "the state of the art" is counting words. *Assessing Writing*, 21:104–111.

Ramineni, C. and Williamson, D. M. (2013). Automated essay scoring: Psychometric guidelines and practices. *Assessing Writing*, 18(1):25–39.

Shermis, M. D. and Hammer, B. (2013). Contrasting state-of-the-art automated scoring of essays. In M. D. Shermis et al., editors, *Handbook of Automated Essay Evaluation*, pages 313–346. Routledge.

Straka, M., Hajič, J., and Straková, J. (2016). UDPipe: Trainable pipeline for processing CoNLL-u files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4290–4297, Portorož, Slovenia. European Language Resources Association (ELRA).

Vajjala, S. and Rama, T. (2018). Experiments with universal CEFR classification. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 147–153, New Orleans, Louisiana. Association for Computational Linguistics.

Weigle, S. C. (2013). English language learners and automated scoring of essays: Critical considerations. *Assessing Writing*, 18(1):85–99.

Williamson, D. M., Xi, X., and Breyer, F. J. (2012). A framework for evaluation and use of automated scoring. *Educational Measurement: Issues and Practices*, 31(1):2–13.

Zhang, D., Li, L., and Zhao, S. (2016). Testing writing in Chinese as a second language: An overview of research. In J. Fox et al., editors, *Trends in language assessment research and practice: The view from the Middle East and the Pacific Rim*, pages 317–335. Cambridge Scholars Publishing.